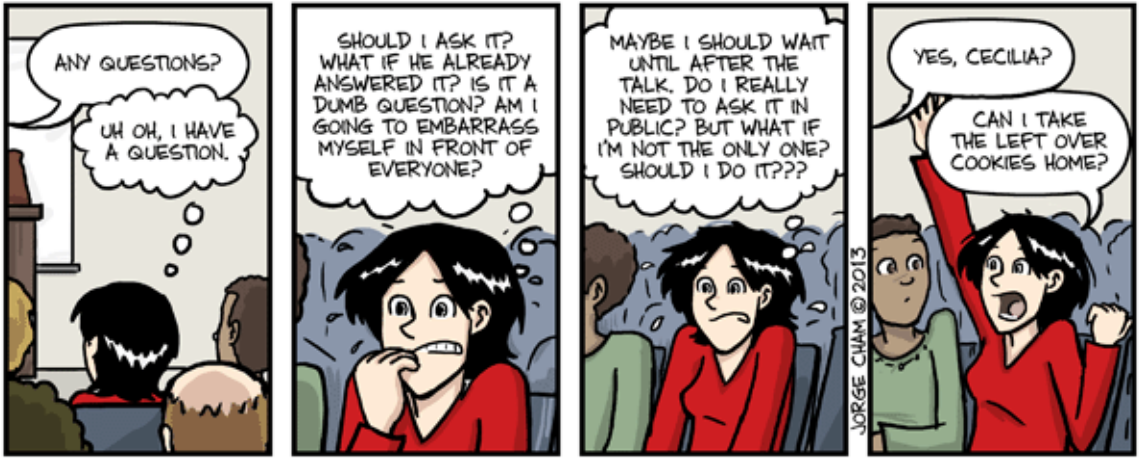
Tensor decompositions and their applications

Lecture 3: Tensor trains decomposition

Nick Vannieuwenhoven (KU Leuven)

1. Introduction (5')
2. Tensor network notation* (45')
   - Definition
   - Examples
   - SVD as a pair of scissors
3. Tensor trains decomposition (35')
4. Application: dynamic tensor approximation (20')
5. Conclusions


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Overview

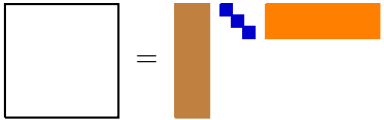
1. Introduction (5')
2. Tensor network notation* (45')
   - Definition
   - Examples
   - SVD as a pair of scissors
3. Tensor trains decomposition (35')
4. Application: dynamic tensor approximation (20')
5. Conclusions

Linear algebra

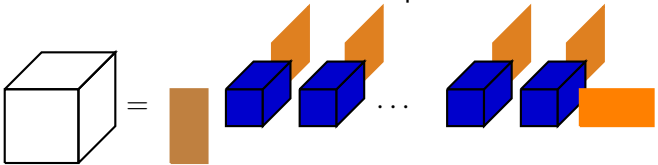
is extended to

Multilinear algebra

Singular value decomposition



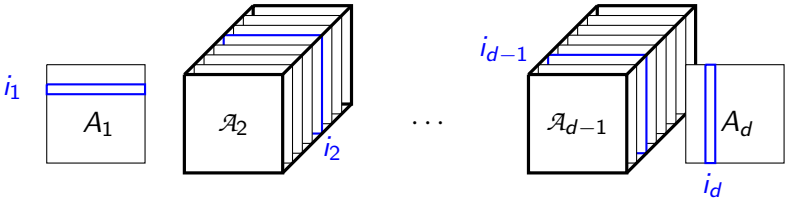
Tensor trains decomposition



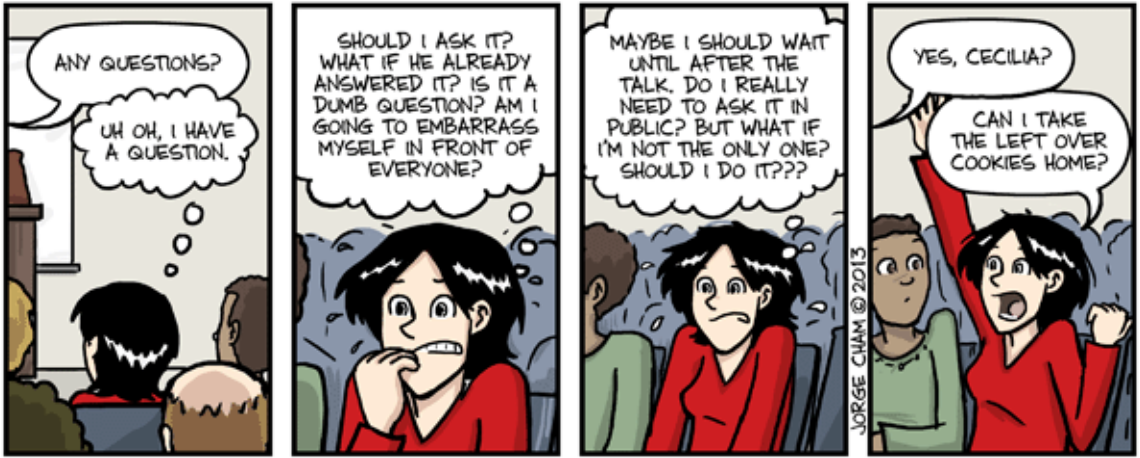
Before we delve into the details, it will be helpful to keep in mind the following **definition of the tensor trains decomposition in coordinates**.

We say that $\mathcal{A} \in \mathbb{k}^{n_1 \times n_2 \times \cdots \times n_d}$ admits a tensor trains decomposition with **bond dimensions** $(r_1, \ldots, r_{d-1})$ if each entry of the tensor is a contracted **matrix chain multiplication**, like so

$$a_{i_1, i_2, \ldots, i_{d-1}, i_d} =$$



In the quantum physics literature where this decomposition originated it is called a **matrix product state**, for obvious reasons!


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

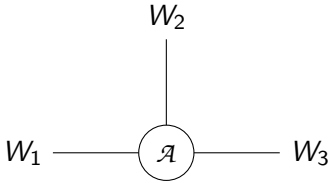
Overview

1. Introduction (5')
2. Tensor network notation* (45')
   - Definition
   - Examples
   - SVD as a pair of scissors
3. Tensor trains decomposition (35')
4. Application: dynamic tensor approximation (20')
5. Conclusions

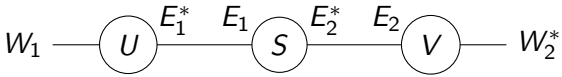
Tensor network notation

In the quantum physics literature a **tensor network notation** originated to succinctly visualize several tensor decompositions. See Ye and Lim (2018) and Orús (2014).

A general third-order tensor $\mathcal{A}$:



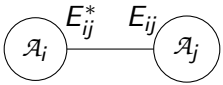
Matrices that have a factorization USV:



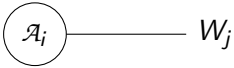
A tensor network (TN) is a visual representation of tensor decompositions that can be realized as **tensor contractions**. A TN consists of the following:

1. Each tensor $\mathcal{A}_i$ in the network is represented by a **node**:

2. Nodes i and j can be connected with **contraction edges**. This is an edge between nodes, labeled on one side with a vector space E_{ij} and on the other side with its dual. The dimension of this vector space is the **bond dimension**.

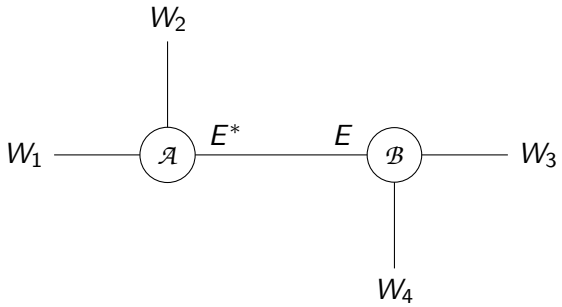


3. Tensors in the network can have **output edges**. This is an edge between a node and an output vector space W_j.



A tensor at a node lives in the tensor product of

1. the output vector spaces of its output edges and
2. one of the vector spaces on the contraction edges.



In the above example,

$$\mathcal{A} \in W_1 \otimes W_2 \otimes E^* \qquad \text{and} \qquad \mathcal{B} \in W_3 \otimes W_4 \otimes E.$$

The tensor represented by a TN is obtained by **contracting along the contraction edges**. Tensors $\mathcal{A}$ and $\mathcal{B}$ in adjacent nodes connected through a vector space E can be contracted to a single tensor, as follows.

Let $\mathcal{A} \in V_1 \otimes \cdots \otimes V_m \otimes E^*$ and $\mathcal{B} \in W_1 \otimes \cdots \otimes W_n \otimes E$. Then, there are expressions

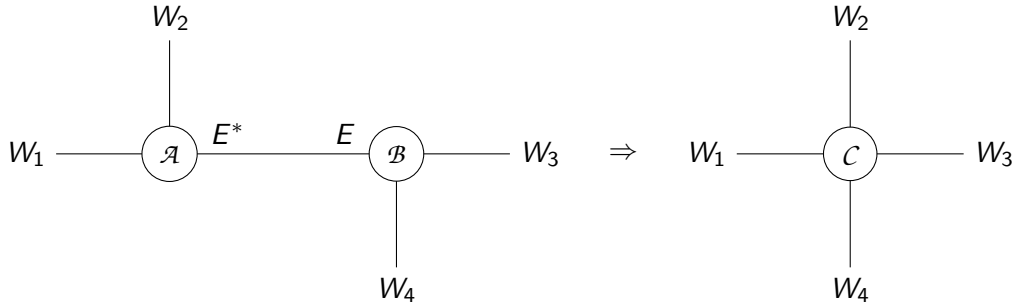
$$\mathcal{A} = \sum_{i=1}^{r} \mathrm{v}_i^1 \otimes \cdots \otimes \mathrm{v}_i^m \otimes \mathrm{f}_i^* \quad \text{and} \quad \mathcal{B} = \sum_{j=1}^{s} \mathrm{w}_j^1 \otimes \cdots \otimes \mathrm{w}_j^n \otimes \mathrm{g}_j.$$

Their **contraction along vector space** E is defined to be the tensor given by tensoring $\mathcal{A}$ and $\mathcal{B}$ and then contracting out E with its dual E^*:

$$\mathcal{C} = \sum_{i=1}^{r} \sum_{j=1}^{s} \mathrm{f}_i^*(\mathrm{g}_j) \cdot \mathrm{v}_i^1 \otimes \cdots \otimes \mathrm{v}_i^m \otimes \mathrm{w}_j^1 \otimes \cdots \otimes \mathrm{w}_j^n;$$

it lives in $V_1 \otimes \cdots \otimes V_m \otimes W_1 \otimes \cdots \otimes W_n$.

Visually, this looks like this:



The tensors

$$\mathcal{A} \in W_1 \otimes W_2 \otimes E^* \quad \text{and} \quad \mathcal{B} \in E \otimes W_3 \otimes W_4$$

can be contracted on E, resulting in $\mathcal{C} \in W_1 \otimes W_2 \otimes V_3 \otimes V_4$.

The left picture generally represents a smaller set of tensors!

Coordinatewise contraction definition

Alternatively, contraction of $\mathcal{A}$ and $\mathcal{B}$ can be defined as follows. Let v_i^k be a basis of V_k, w_i^k a basis of W_k, and e_i a basis of E with dual basis e_i^*. Then,

$$\mathcal{A} = [a_{i_1,\ldots,i_m,j}] = \sum_{i_1=1}^{\dim V_1} \cdots \sum_{i_m=1}^{\dim V_m} \sum_{j=1}^{\dim E^*} a_{i_1,\ldots,i_m,j} \mathrm{v}_{i_1}^1 \otimes \cdots \otimes \mathrm{v}_{i_m}^m \otimes \mathrm{e}_j^*$$

$$\mathcal{B} = \left[b_{j_1,\ldots,j_m,j'}\right] = \sum_{j_1=1}^{\dim W_1} \cdots \sum_{j_n=1}^{\dim W_n} \sum_{j'=1}^{\dim E} b_{j_1,\ldots,j_n,j'} \mathrm{w}_{j_1}^1 \otimes \cdots \otimes \mathrm{w}_{j_n}^n \otimes \mathrm{e}_{j'}$$

and the contraction of $\mathcal{A}$ and $\mathcal{B}$ is

Coordinatewise contraction definition

Alternatively, contraction of $\mathcal{A}$ and $\mathcal{B}$ can be defined as follows. Let v_i^k be a basis of V_k, w_i^k a basis of W_k, and e_i a basis of E with dual basis e_i^*. Then,

$$\mathcal{A} = [a_{i_1,\ldots,i_m,j}] = \sum_{i_1=1}^{\dim V_1} \cdots \sum_{i_m=1}^{\dim V_m} \sum_{j=1}^{\dim E^*} a_{i_1,\ldots,i_m,j} \mathrm{v}_{i_1}^1 \otimes \cdots \otimes \mathrm{v}_{i_m}^m \otimes \mathrm{e}_j^*$$

$$\mathcal{B} = \left[b_{j_1,\ldots,j_m,j'}\right] = \sum_{j_1=1}^{\dim W_1} \cdots \sum_{j_n=1}^{\dim W_n} \sum_{j'=1}^{\dim E} b_{j_1,\ldots,j_n,j'} \mathrm{w}_{j_1}^1 \otimes \cdots \otimes \mathrm{w}_{j_n}^n \otimes \mathrm{e}_{j'}$$

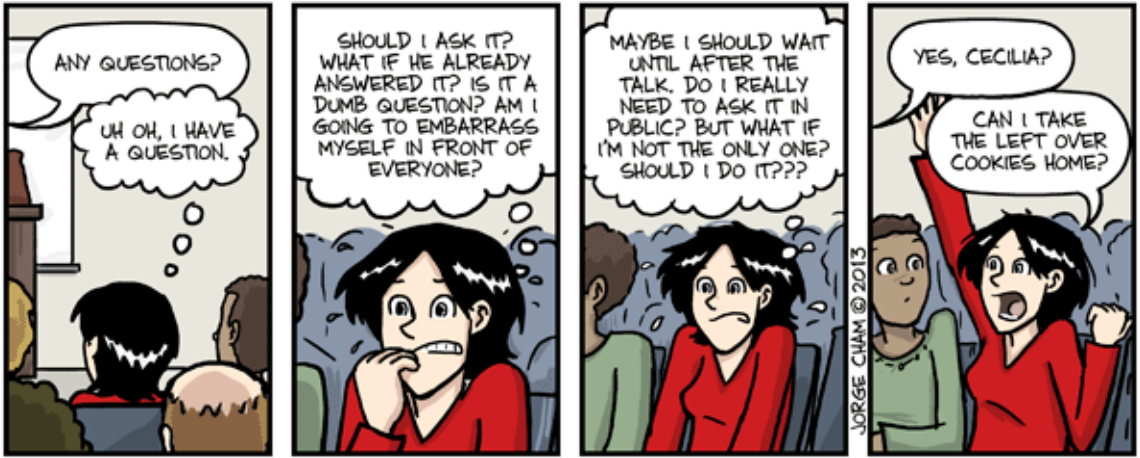
and the contraction of $\mathcal{A}$ and $\mathcal{B}$ is

$$\mathcal{C} = \sum_{i_1=1}^{\dim V_1} \cdots \sum_{i_m=1}^{\dim V_m} \sum_{j_1=1}^{\dim W_1} \cdots \sum_{j_n=1}^{\dim W_n} \sum_{j=1}^{\dim E} \sum_{j'=1}^{\dim E} \delta_{jj'} a_{i_1,\ldots,i_m,j} b_{j_1,\ldots,j_n,j'} \mathrm{v}_{i_1}^1 \otimes \cdots \otimes \mathrm{v}_{i_m}^m \otimes \mathrm{w}_{j_1}^1 \otimes \cdots \otimes \mathrm{w}_{j_n}^n$$

$$= \left[\sum_{j=1}^{\dim E} a_{i_1,\ldots,i_m,j} b_{j_1,\ldots,j_n,j} \right]$$

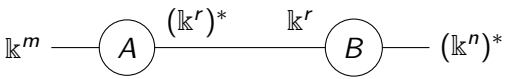
In conclusion, an edge between two tensors can be **contracted in coordinates** by summing over the joint index that the edge represents.

The **key point** is that this is true for any choice of coordinates!


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Example I: Low-rank matrix decomposition

Consider the concrete example:



Let e_i be basis vectors of $\mathbb{k}^m$, f_j basis vectors of $\mathbb{k}^r$ (with dual basis vectors f_j^*), and g_k^* basis vectors of $(\mathbb{k}^n)^*$. Let

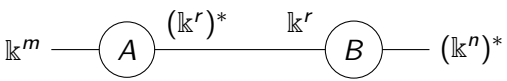
$$A = [a_{ij}]_{ij} = \sum_{i=1}^{m} \sum_{j=1}^{r} a_{ij} e_i \otimes f_j^* \quad \text{and} \quad B = [b_{j'k}]_{j'k} = \sum_{j'=1}^{r} \sum_{k=1}^{n} b_{j'k} f_{j'} \otimes g_k^*.$$

Then,

$$C = \left[\sum_{j=1}^{r} a_{ij} b_{jk} \right]_{ik} = \sum_{i=1}^{m} \sum_{k=1}^{n} \left(\sum_{j=1}^{r} a_{ij} b_{jk} \right) e_i \otimes g_k^*$$

Example I: Low-rank matrix decomposition

Consider the concrete example:

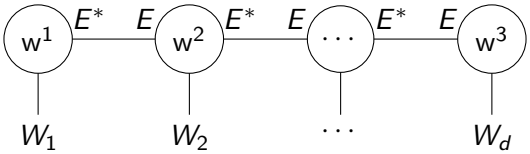


This represents the **matrix multiplication** of $A \in \mathbb{k}^{m \times r}$ with $B \in \mathbb{k}^{r \times n}$. The matrices in $\mathbb{k}^{m \times n}$ represented by this tensor network are all of the form $C = AB$.

When $r \leq m, n$, this can be interpreted as a **low-rank matrix decomposition**.

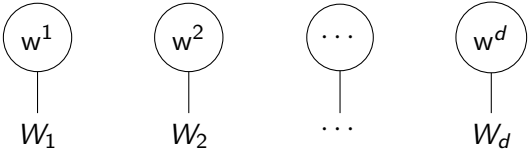
Example II: rank-1 tensors

A rank-1 tensor $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ is the tensor product of vectors in W_i. Let E be a one-dimensional vector space with basis vector e and dual basis e^*. Then the TN notation for a rank-1 tensor is:



$$\mathcal{A} = (e^*(e) \cdots e^*(e))\, \mathrm{w}^1 \otimes \mathrm{w}^2 \otimes \cdots \otimes \mathrm{w}^d = \mathrm{w}^1 \otimes \mathrm{w}^2 \otimes \cdots \otimes \mathrm{w}^d$$

This is usually abbreviated to

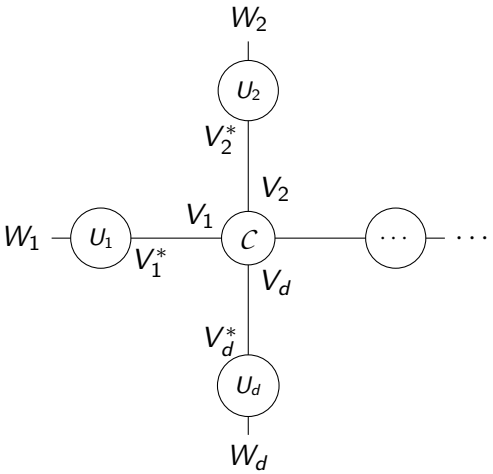


Example III: Tucker decomposition, third interpretation

Tensors admitting a Tucker decomposition with multilinear rank bounded by $(r_1, \ldots, r_d)$ are represented by the TN on the right, assuming $r_i = \dim V_i$.

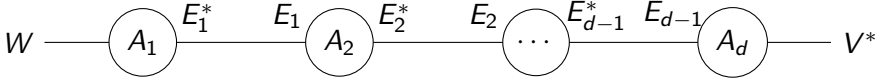
After some computations, you can find that the tensors represented by this network are of the form

$$\mathcal{A} = \left[\sum_{j_1=1}^{r_1} \cdots \sum_{j_d=1}^{r_d} (U_1)_{i_1,j_1} \cdots (U_d)_{i_d,j_d} c_{j_1,\ldots,j_d} \right]_{i_1,\ldots,i_d}$$



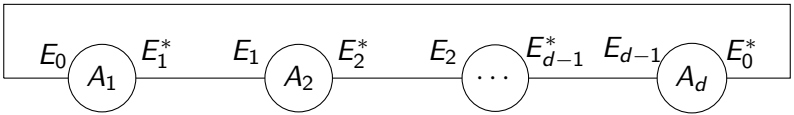
Example IV: Matrix chain and cycle multiplication

Matrix chain decomposition:



$$A = A_1 A_2 \cdots A_d$$

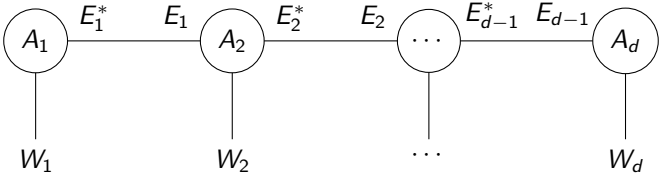
Matrix cycle decomposition:



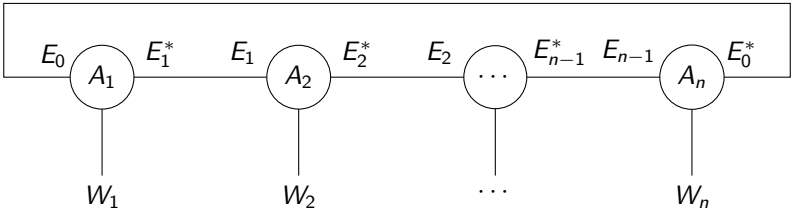
$$a = \mathrm{trace}(A_1 A_2 \cdots A_d)$$

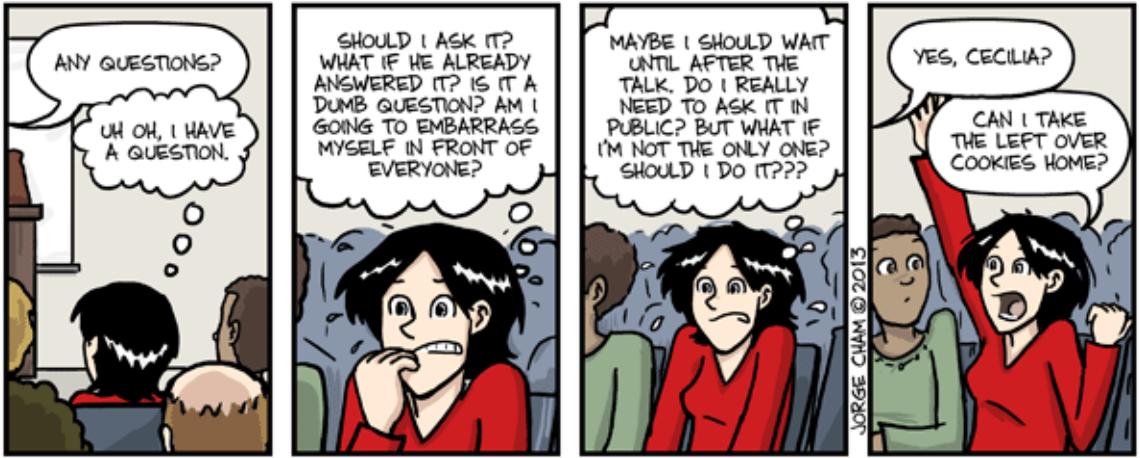
Example V: Matrix product states

Tensor trains decomposition:



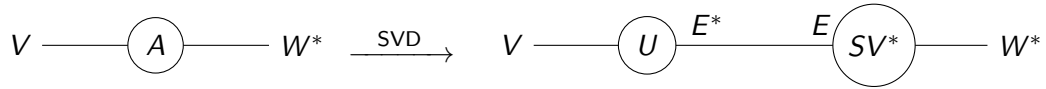
Tensor ring decomposition:



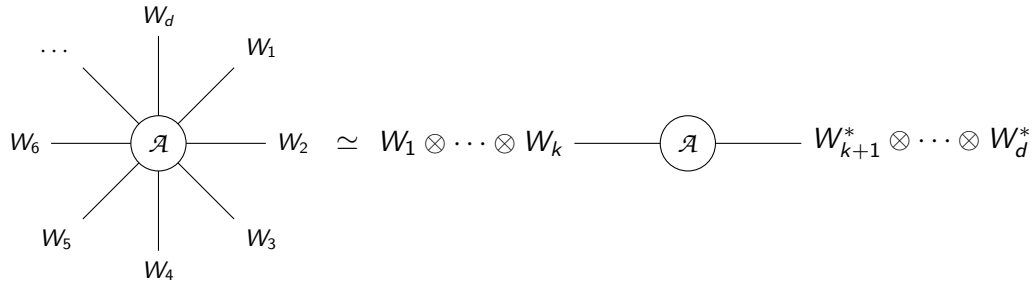

ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

The SVD as a pair of scissors

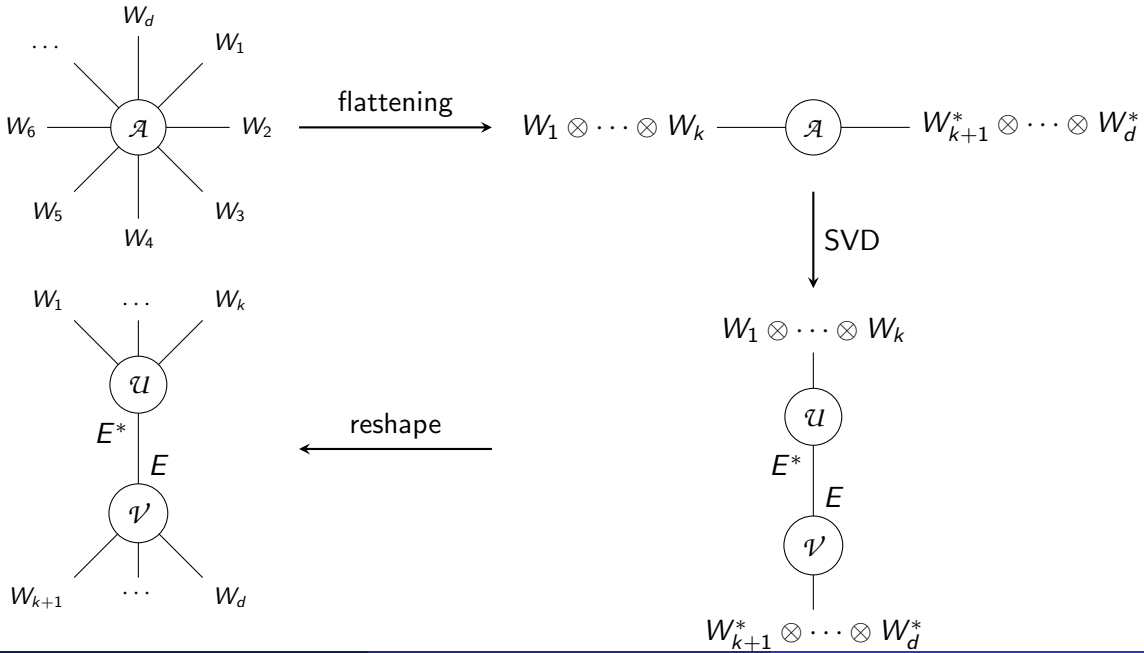
The singular value decomposition is a **way of cutting up the tensor product of vector spaces** into "smaller" pieces. For two vector spaces we have:

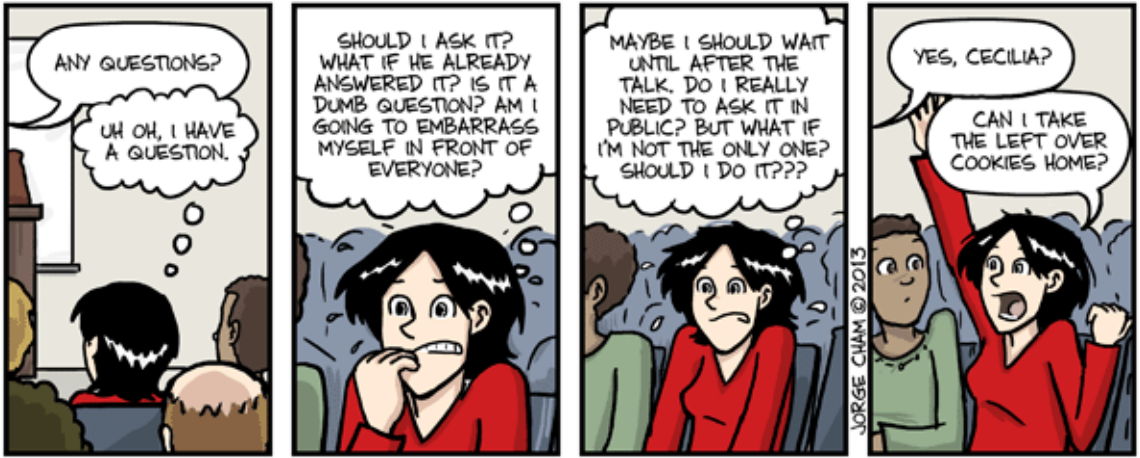


Combining this with flattenings



creates a very potent tool!


W_d
W_1
W_6
$\mathcal{A}$
W_2
W_5
W_3
W_4
flattening
$W_1 \otimes \cdots \otimes W_k$
$\mathcal{A}$
$W_{k+1}^* \otimes \cdots \otimes W_d^*$
SVD
$W_1 \otimes \cdots \otimes W_k$
$\mathcal{U}$
E^*
E
$\mathcal{V}$
$W_{k+1}^* \otimes \cdots \otimes W_d^*$
reshape
W_1
W_k
$\mathcal{U}$
E^*
E
$\mathcal{V}$
W_{k+1}
W_d

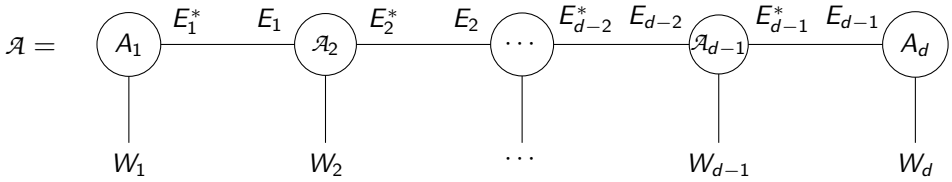

ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Overview

1. Introduction (5')
2. Tensor network notation* (45')
   - Definition
   - Examples
   - SVD as a pair of scissors
3. Tensor trains decomposition (35')
4. Application: dynamic tensor approximation (20')
5. Conclusions

Tensor trains decomposition: first interpretation

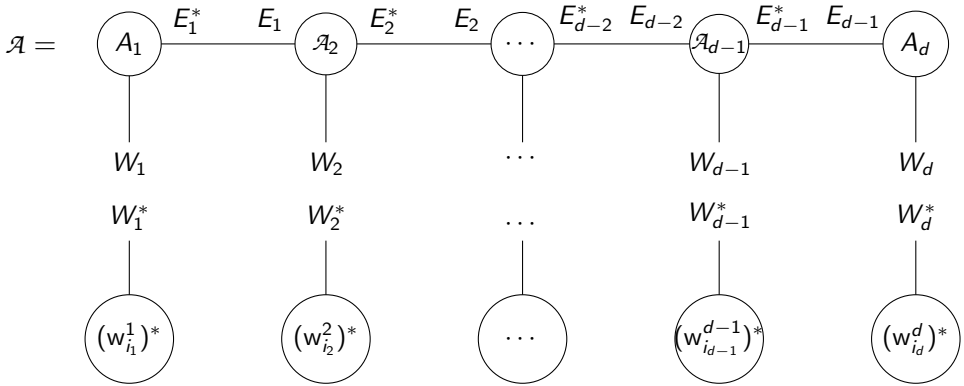
A **tensor trains decomposition** of a tensor $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ with bond dimensions $(r_1, \ldots, r_{d-1})$ is a TN with the following structure:



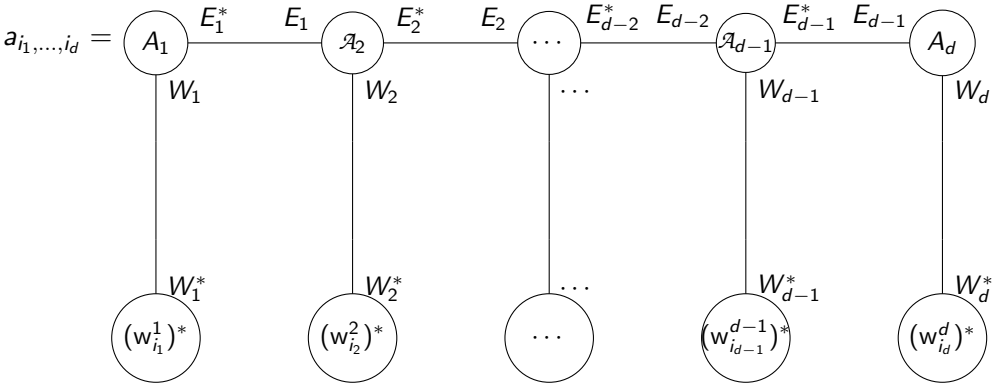
where $r_i = \dim E_i$.

Let's try to view this in coordinates ...

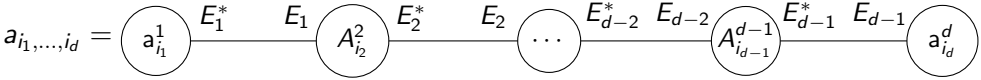
Let w_i^k be a basis of W_k, and assume that $\mathcal{A}$ is expressed relative to the tensor product basis. Contracting $\mathcal{A}$ with a basis vector $\mathrm{w}_{i_1}^1 \otimes \cdots \otimes \mathrm{w}_{i_d}^d$, we get:



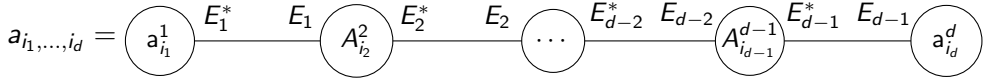
Let w_i^k be a basis of W_k, and assume that $\mathcal{A}$ is expressed relative to the tensor product basis. Contracting $\mathcal{A}$ with a basis vector $\mathrm{w}_{i_1}^1 \otimes \cdots \otimes \mathrm{w}_{i_d}^d$, we get:



Let w_i^k be a basis of W_k, and assume that $\mathcal{A}$ is expressed relative to the tensor product basis. Contracting $\mathcal{A}$ with a basis vector $\mathrm{w}_{i_1}^1 \otimes \cdots \otimes \mathrm{w}_{i_d}^d$, we get:



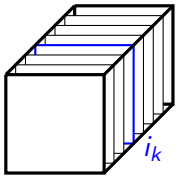
Let w_i^k be a basis of W_k, and assume that $\mathcal{A}$ is expressed relative to the tensor product basis. Contracting $\mathcal{A}$ with a basis vector $\mathrm{w}_{i_1}^1 \otimes \cdots \otimes \mathrm{w}_{i_d}^d$, we get:



$$= (\mathrm{a}_{i_1}^1)^T A_{i_2}^2 A_{i_3}^3 \cdots A_{i_{d-1}}^{d-1} \mathrm{a}_{i_d}^d$$

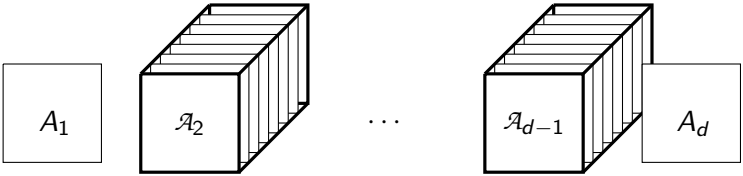
where $A_{i_k}^k$ is the i_kth slice of $\mathcal{A}_k \in E_{k-1} \otimes E_k^* \otimes W_k$ in the factor W_k; that is,

$$A_{i_k}^k = (I, I, \mathrm{w}_{i_k}^k)^* \cdot \mathcal{A}_k =$$



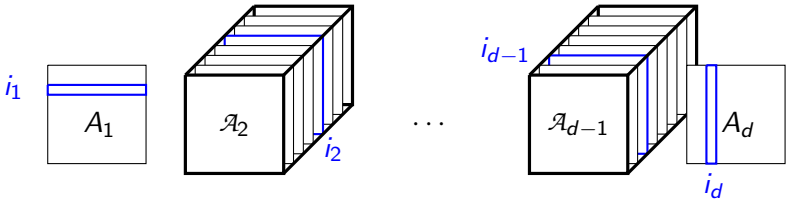
In conclusion, a tensor $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ admits a tensor trains decomposition if there exists a tuple

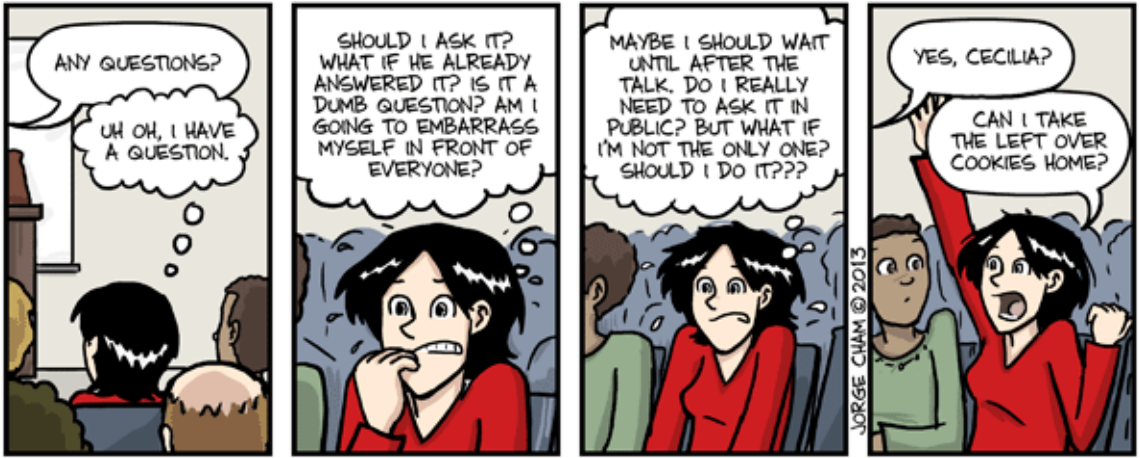
$$(A_1, \mathcal{A}_2, \ldots, \mathcal{A}_{d-1}, A_d) =$$



such that

$$a_{i_1,i_2,\ldots,i_{d-1},i_d} =$$




ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Tensor trains decomposition: second interpretation

A complementary way to view a tensor trains decomposition by Grasedyck (2010) comes from **successively nested partially separable tensor product subspaces**. The tensor trains assumption is that a tensor $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$ lives in the **intersection** of the following **successively nested subspaces**

$$
\begin{aligned}
&V_1 \otimes W_2 \otimes \cdots \otimes W_d && \\
&V_{1,2} \otimes W_3 \otimes \cdots \otimes W_d && V_{1,2} \subset V_1 \otimes W_2 \\
&V_{1,2,3} \otimes W_4 \otimes \cdots \otimes W_d && V_{1,2,3} \subset V_{1,2} \otimes W_3 \\
&\qquad \vdots && \qquad \vdots \\
&V_{1,\ldots,d-1} \otimes W_d && V_{1,\ldots,d-1} \subset V_{1,\ldots,d-2} \otimes W_{d-1}
\end{aligned}
$$

That is, a tensor $\mathcal{A}$ living in the intersection of the above vector spaces has a tensor trains decomposition with bond dimensions $(\dim V_1, \dim V_{1,2}, \ldots, \dim V_{1,\ldots,d-1})$.

Assume $\mathcal{A} = [a_{i_1,\ldots,i_d}]$ is expressed with respect to the tensor product of the bases $W_k = [\mathrm{w}_i^k]$ for W_k. Let $U_{1,\ldots,k}$ be a basis of $V_{1,\ldots,k}$ (expressed w.r.t. the tensor product basis $W_1 \otimes \cdots \otimes W_k$). Denote

$$n_k = \dim W_k \qquad \text{and} \qquad r_k = \dim V_{1,\ldots,k}.$$

Since $V_{1,\ldots,k} \subset V_{1,\ldots,k-1} \otimes W_k$ there exists a matrix $A_k \in \mathbb{k}^{r_{k-1} \cdot n_k \times r_k}$ such that

$$U_{1,\ldots,k} = (U_{1,\ldots,k-1} \otimes W_k) A_k,$$

and because $\mathcal{A} \in V_{1,\ldots,d-1} \otimes W_d$, there also exists a final matrix $A_d \in \mathbb{k}^{r_{d-1} \times n_d}$ such that

$$\mathcal{A}_{(1,\ldots,d-1;d)} = U_{1,\ldots,d-1} A_d.$$

After recursively unwinding the foregoing definitions, we have

$$\mathcal{A}_{(1,\ldots,d-1;d)} = (((U_1 \otimes W_2)A_2 \otimes W_3)\, A_3 \cdots \otimes W_{d-1})\, A_{d-1}A_d$$

We can now compute

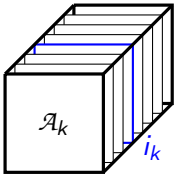
$$\begin{aligned} a_{i_1,\ldots,i_d} &= (\mathrm{w}_{i_1}^1, \ldots, \mathrm{w}_{i_d}^d)^* \cdot \mathcal{A} \\ &= (\mathrm{w}_{i_1}^1 \otimes \cdots \otimes \mathrm{w}_{i_{d-1}}^{d-1})^* \mathcal{A}_{(1,\ldots,d-1;d)} \mathrm{w}_{i_d}^d \\ &= \left((((\mathrm{w}_{i_1}^1)^* U_1 \otimes (\mathrm{w}_{i_2}^2)^* W_2)A_2 \otimes (\mathrm{w}_{i_3}^3)^* W_3)\, A_3 \cdots \otimes (\mathrm{w}_{i_{d-1}}^{d-1})^* W_{d-1} \right) A_{d-1}A_d \mathrm{w}_{i_d}^d. \end{aligned}$$

We observe that, suitably defining $\mathcal{A}_k \in \Bbbk^{r_{k-1} \times r_k \times n_k}$, we have for every compatible X, Y that

$$(X \otimes Y)A_k = (X \otimes Y)(\mathcal{A}_k)_{(1,3;2)} = ((X, I, Y) \cdot \mathcal{A}_k)_{(1,3;2)}$$

Note that $(\mathrm{w}_{i_k}^k)^* W_k$ vanishes completely, except on the i_kth basis vector where it is 1. Consequently,

$$(I \otimes (\mathrm{w}_{i_k}^k)^* W_k) A_k = e_{i_k}^T \cdot_3 \mathcal{A}_k =$$

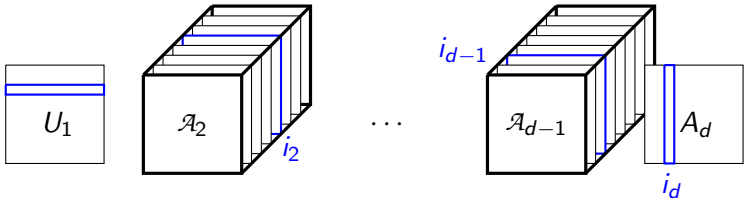


Hence, further parsing our expression, we find

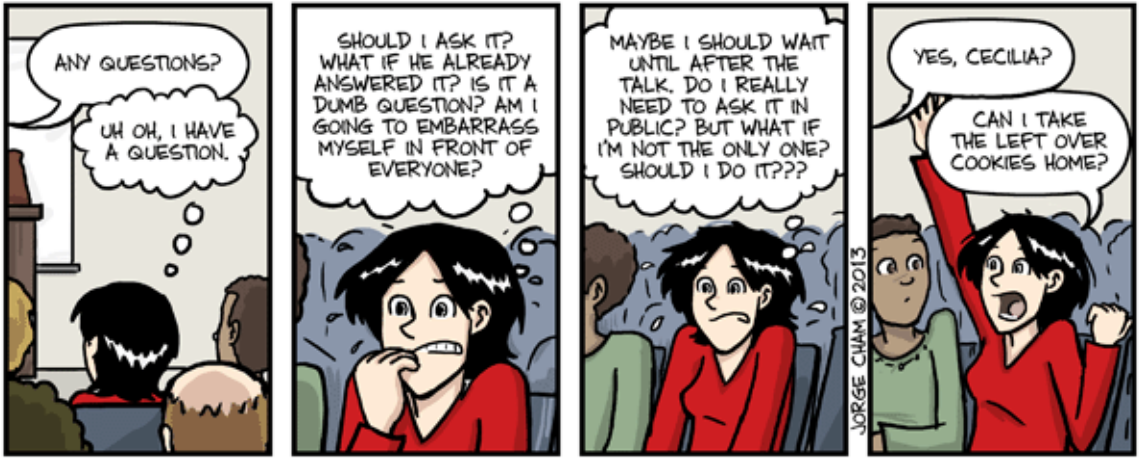
$$\begin{aligned} a_{i_1,\ldots,i_d} &= \left((((\mathrm{w}_{i_1}^1)^* U_1 \otimes (\mathrm{w}_{i_2}^2)^* W_2) A_2 \otimes (\mathrm{w}_{i_3}^3)^* W_3) \, A_3 \cdots \otimes (\mathrm{w}_{i_{d-1}}^{d-1})^* W_{d-1} \right) A_{d-1} A_d \mathrm{w}_{i_d}^d \\ &= (U_1)_{i_1,:} \, (e_{i_2}^T \cdot_3 \mathcal{A}_2)(e_{i_3}^T \cdot_3 \mathcal{A}_3) \cdots (e_{i_{d-1}}^T \cdot_3 \mathcal{A}_{d-1}) \, (A_d)_{:,i_d} \end{aligned}$$

In conclusion, we have

$$a_{i_1,i_2,\ldots,i_{d-1},i_d} =$$



$$= (U_1)_{i_1,:} \, (e_{i_2}^T \cdot_3 \mathcal{A}_2)(e_{i_3}^T \cdot_3 \mathcal{A}_3) \cdots (e_{i_{d-1}}^T \cdot_3 \mathcal{A}_{d-1}) \, (A_d)_{:,i_d}$$


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Tensor trains rank

Observe that if

$$\mathcal{A} \in V_{1,\dots,k} \otimes W_{k+1} \otimes \cdots \otimes W_d \quad \text{with} \quad V_{1,\dots,k} \subset W_1 \otimes \cdots \otimes W_k$$

then necessarily

$$\mathcal{A}_{(1,\dots,k;k+1,\dots,d)} \in V_{1,\dots,k} \otimes (W_{k+1} \otimes \cdots \otimes W_d)^*$$

so

$$V'_{1,\dots,k} = \operatorname{span}(\mathcal{A}_{(1,\dots,k;k+1,\dots,d)}) \subset V_{1,\dots,k}.$$

In fact, choosing $V'_{1,\dots,k}$ results in the minimal nested partially separable tensor product subspace in which $\mathcal{A}$ lives.

The corresponding **tensor trains rank** is defined to be

$$\left(\dim V'_1, \dim V'_{1,2}, \dots, \dim V'_{1,\dots,d-1}\right).$$

The tensor trains decomposition provides a **data sparse representation** of tensors $\mathcal{A}$ living in successively nested subspaces.

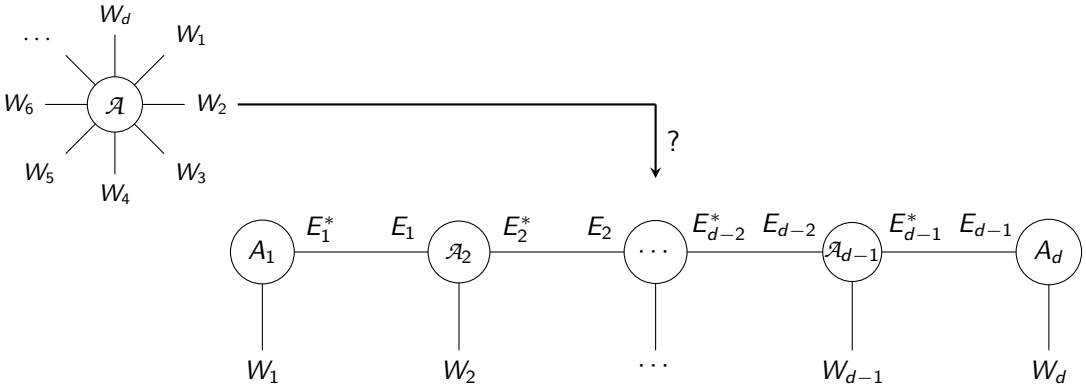
If $\mathcal{A} \in \mathbb{k}^{n_1 \times n_2 \times \cdots \times n_d}$ has tensor trains rank $(r_1, r_2, \ldots, r_{d-1})$, then it can be represented exactly using only

$$\underbrace{n_1 r_1}_{\text{first matrix}} + \underbrace{r_{d-1} n_d}_{\text{final matrix}} + \sum_{i=2}^{d-1} \underbrace{r_{k-1} r_k n_k}_{\text{transfer tensors}}$$

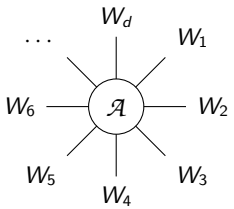
storage (for A_1, A_d and the $\mathcal{A}_i$'s).

Computing a minimal tensor trains decomposition

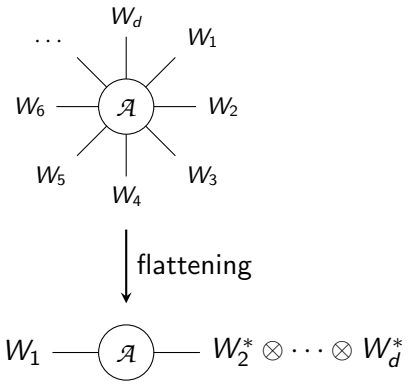
Using compact SVD as our scissors, how do we do this?



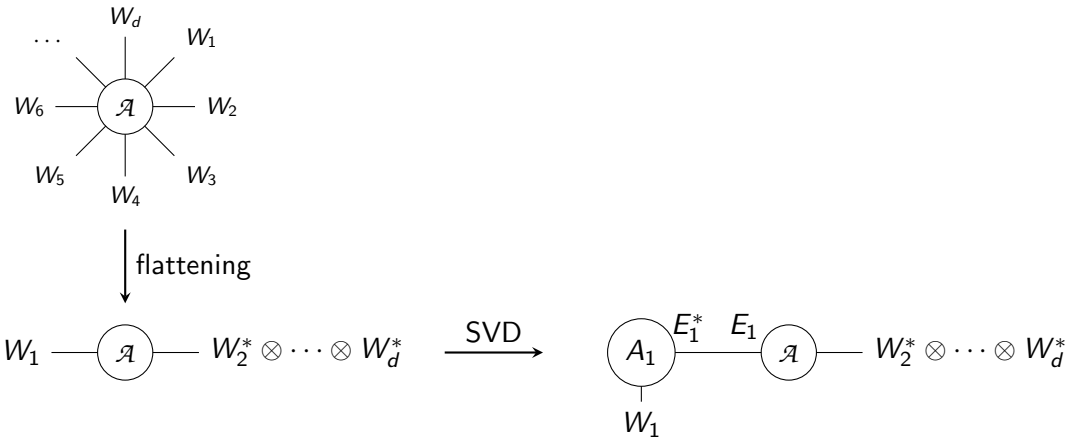
Here's how it goes:



Here's how it goes:

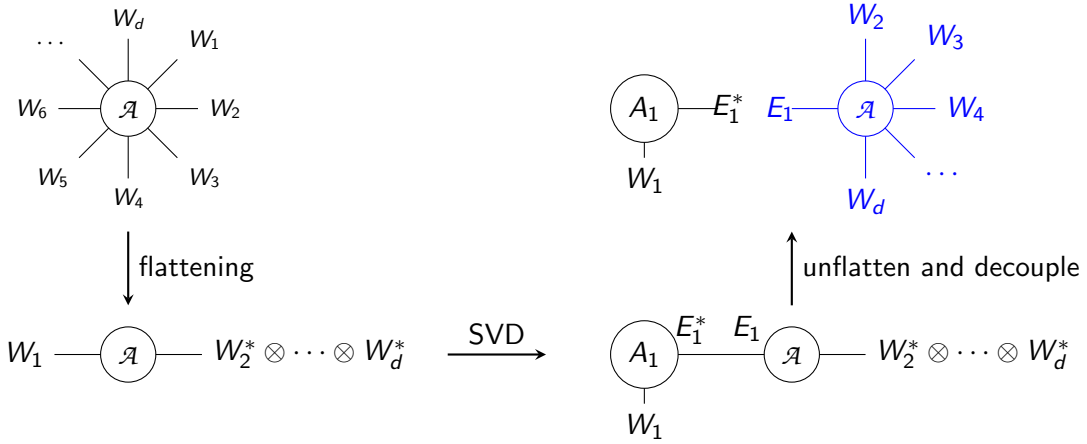


Here's how it goes:

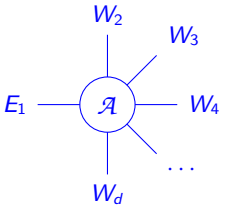


For data tensors, replace the compact SVD by a truncated SVD. This will give you an optimal approximation each time you apply the scissors.

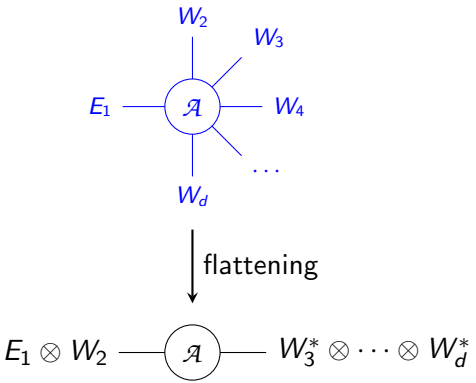
Here's how it goes:



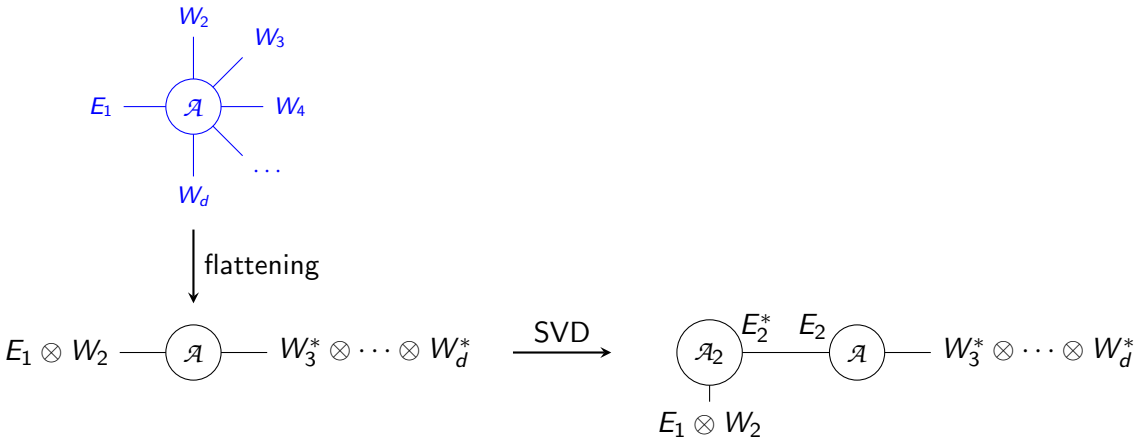
Here's how it goes:



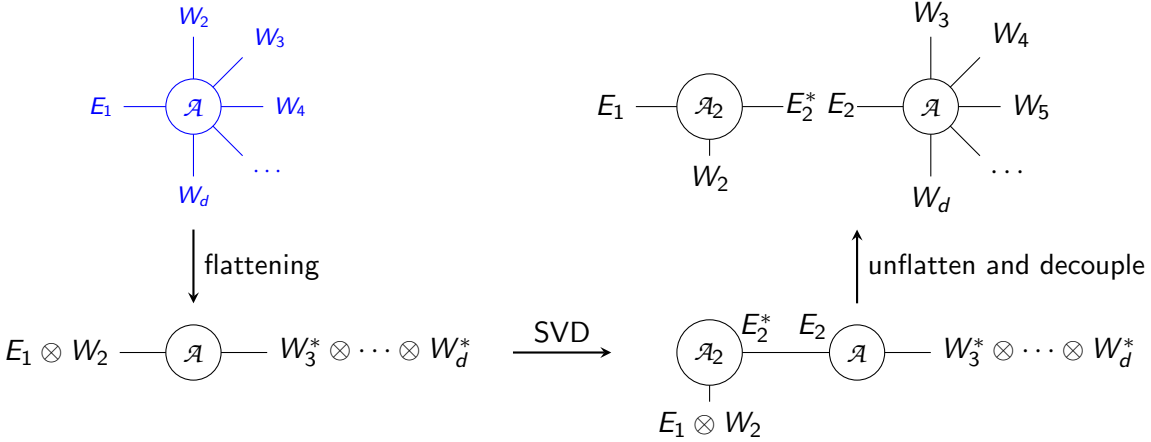
Here's how it goes:



Here's how it goes:



Here's how it goes:



TT-SVD algorithm

Algorithm 1: TT-SVD Algorithm (Oseledets, 2011)

input : A tensor $\mathcal{A} \in \mathbb{k}^{n_1 \times n_2 \times \cdots \times n_d}$

input : A target tensor trains rank $(r_1, r_2, \ldots, r_{d-1})$.

output: The components $(A_1, \mathcal{A}_2, \ldots, \mathcal{A}_{d-1}, A_d)$ of the tensor trains decomposition

Compute the rank-r_1 truncated SVD $\mathcal{A}_{(1;2,\ldots,d)} \approx A_1 \Sigma_1 Q_1^*$;

$\mathcal{A}_{(1;2,\ldots,d-1)} \leftarrow \Sigma_1 Q_1^*$;

for $k = 2, \ldots, d-1$ **do**

  Compute the rank-r_k truncated SVD $\mathcal{A}_{(1,2;3,\ldots,d-k+1)} \approx U_k \Sigma_k Q_k^*$;

  $(\mathcal{A}_k)_{(1,3;2)} \leftarrow U_k$;

  $\mathcal{A}_{(1;2,\ldots,d-k)} \leftarrow \Sigma_k Q_k^*$;

end

$A_d \leftarrow \mathcal{A}_{(1;2)}$;

Note: the only difference with the ST-HOSVD is using slightly different flattenings.

Approximation error

The approximation produced by TT-SVD is **quasi-optimal**.

Proposition (Oseledets, 2011)

Let $\mathcal{A} \in W_1 \otimes \cdots \otimes W_d$, *and let* $\mathcal{A}^*$ *be the best tensor trains rank-*$(r_1, \ldots, r_{d-1})$ *approximation to* $\mathcal{B}$, *i.e.,*

$$\|\mathcal{A} - \mathcal{A}^*\|_F = \min_{\text{ttrank}(\mathcal{B}) \leq (r_1, \ldots, r_{d-1})} \|\mathcal{A} - \mathcal{B}\|_F.$$

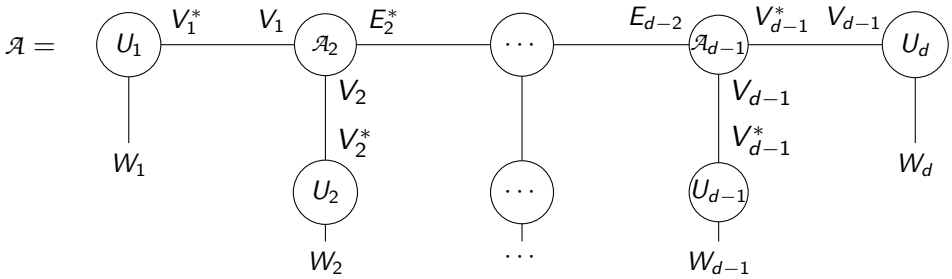
Then, the tensor trains rank-$(r_1, \ldots, r_{d-1})$ *TT-SVD approximation* $\mathcal{A}_{TT}$ *is a* quasi-best approximation*:*

$$\|\mathcal{A} - \mathcal{A}_{TT}\|_F \leq \sqrt{d-1}\|\mathcal{A} - \mathcal{A}^*\|_F.$$

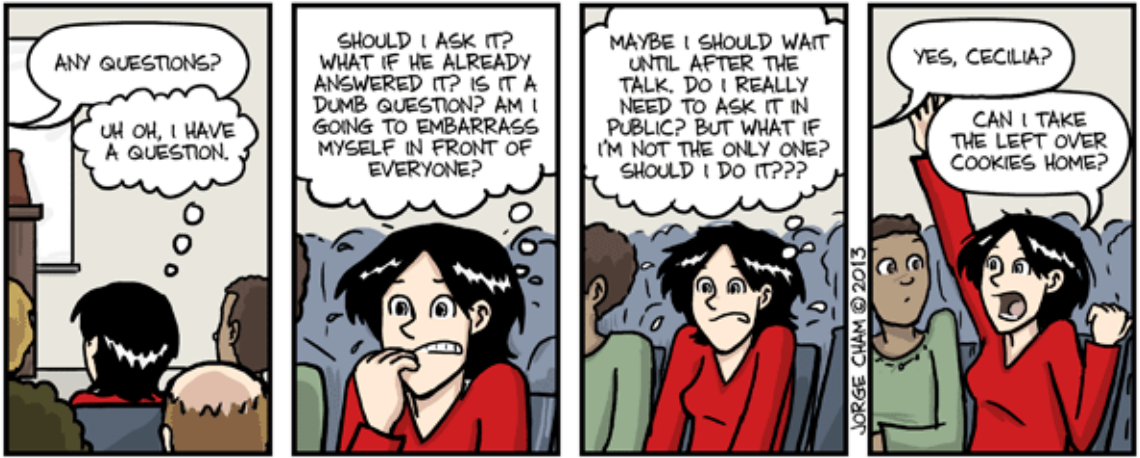
Subspace-revealing variant

The tensor trains decomposition can be further refined to reveal a tensor product basis of the minimal separable tensor subspace in which $\mathcal{A}$ lives.

For this it suffices to compute one more SVD for each $\mathcal{A}_i$ in the tensor trains:



By a suitable change of basis along the contraction edges, all U_i can be chosen as matrices with orthonormal columns.


ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Overview

1. Introduction (5')
2. Tensor network notation* (45')
   - Definition
   - Examples
   - SVD as a pair of scissors
3. Tensor trains decomposition (35')
4. Application: dynamic tensor approximation (20')
5. Conclusions

Application: dynamic tensor approximation

A major advantage of the tensor trains decomposition is that it allows fast operations, like

1. **scalar multiplication** consists of multiplying any of the tensor or matrices by the scalar;
2. **multilinear multiplication** [what does it look like in tensor network notation?];
3. **recompression**, essentially by applying HOSVD compression to the transfer tensors in the tensor train;
4. **adding two tensor trains** with bond dimensions $(r_1, \ldots, r_{d-1})$ and $(s_1, \ldots, s_{d-1})$ can be implemented reasonably, but requires recompression as bond dimensions grow to $(r_1 + s_1, \ldots, r_{d-1} + s_{d-1})$.

Tensor trains are a great tool for approximating and compactly representing **smooth and tensorized functions** (discretizing either the domain or function space).

Large-scale tensor train approximations have been used for

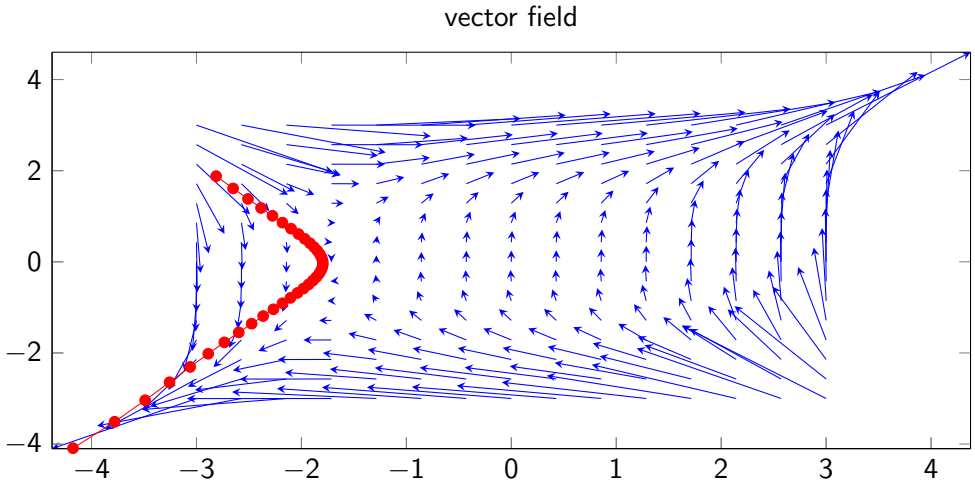
- approximating the ground state of 1D **quantum many-body systems** (White, 1992);
- approximately solving **linear systems** originating from differential equations (Oseledets and Dolgov, 2012);
- approximately solving **partial differential equations** (Dolgov, Khoromskij and Oseledets, 2012);
- approximate **integration of time-dependent partial differential equations** on tensor grids (Lubich, Rohwedder, Schneider and Vandereycken, 2013)
- approximate solution of **stochastic partial differential equations** (Dolgov, Khoromskij, Litvinenko and Matthies, 2015)
- **tensor completion** for incomplete data (Grasedyck, Kluge and Krämer, 2015);
- dimensionality reduction in **deep neural networks** (Novikov, Podoprikhin, Osokin and Vetrov, 2015);
- approximately solving **parameterized eigenvalue problems** (Ruymbeek, Meerbergen and Michiels, 2020);

Consider solving a **time-dependent partial differential equation**

$$\frac{\partial}{\partial t} u(x,t) = g(u(x,t))$$

where $u(x,t)$ is the unknown solution, $g(y)$ is a nonlinear operator, and the initial value $u(x,0)$ is given.

Abstractly, this differential equation defines a **vector field** in (x, t) that should be integrated in the time direction:



Assume that we solve this differential equation with a **forward integrator**, e.g., forward Euler. That is, the solution $u(x, s)$ is approximated by $U_s(x)$ with the scheme

$$U_{s+\epsilon}(x) \leftarrow U_s(x) + \epsilon \cdot \underbrace{g(U_s(x))}_{\text{tangent direction}}$$

Suppose that at each timestep s, the solution $u(x, s)$ can be approximated well by a tensor trains decomposition with fixed bond dimensions. How can we exploit this?

Option 1: Ad-hoc approximation

Naively, we could add an approximation step

$$\hat{U}_{s+\epsilon}(x) \leftarrow \mathrm{App}\left(\hat{U}_s(x) + \epsilon \cdot g(\hat{U}_s(x))\right)$$

where App takes an element from the ambient space and finds the closest tensor trains approximation with fixed bond dimensions.

Finding the best approximation cannot be guaranteed generally. A more practical scheme is

$$\hat{U}_{s+\epsilon}(x) \leftarrow \text{TT-SVD}\left(\hat{U}_s(x) + \epsilon \cdot g(\hat{U}_s(x))\right)$$

where TT-SVD applies the **TT-SVD algorithm** with some fixed bond dimensions.

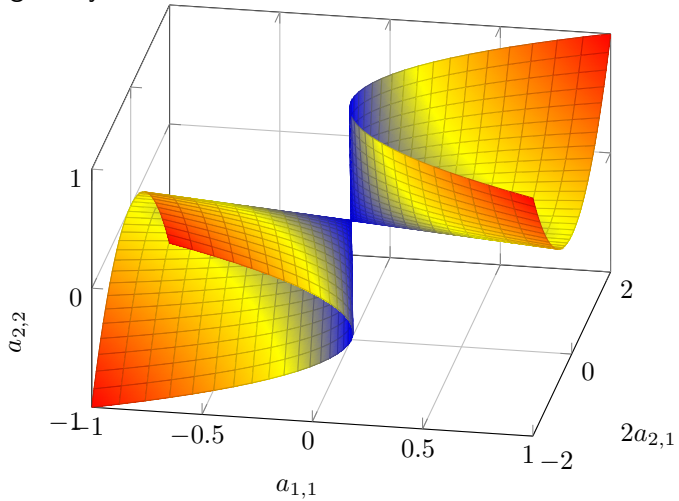
Option 2: Integration on manifolds

An alternative consists of exploiting the following result.

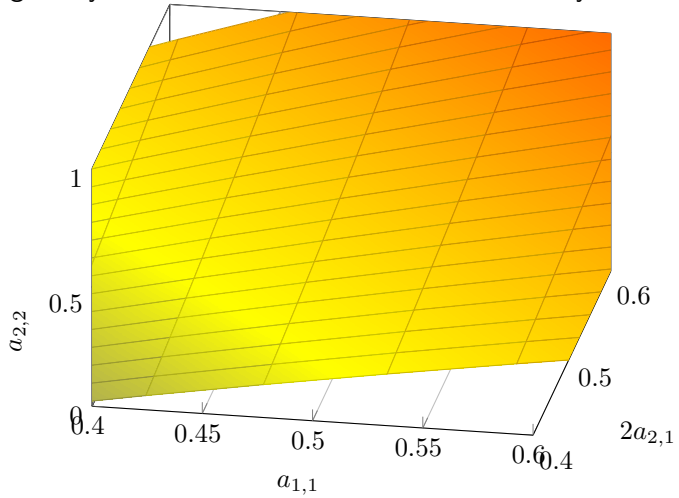
Theorem (Holtz, Rohwedder and Schneider, 2012)

The set of all tensor train decompositions with bond dimension equal to $(r_1, \ldots, r_{d-1})$ *has the structure of a* smooth submanifold *of* $\mathbb{R}^{n_1 \times \cdots \times n_d}$.

A smooth manifold globally looks like a smooth surface



A smooth manifold globally looks like a smooth surface, and locally like a linear space



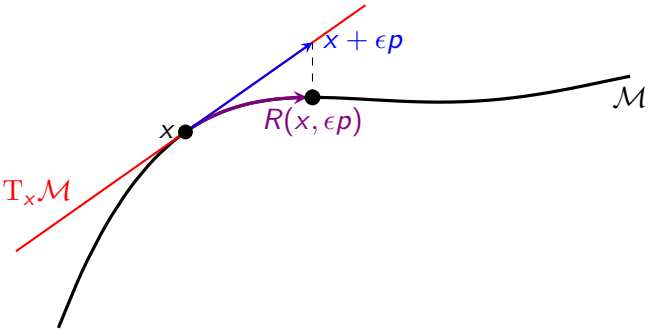
At every point x of a manifold $\mathcal{M}$, this first-order approximation of $\mathcal{M}$ at x is called the **tangent space** $\mathrm{T}_x\mathcal{M}$. The dimension of this space measures the size (dimension) of $\mathcal{M}$.

Assuming we want a solution on the manifold of tensor train decompositions $\mathcal{T}$, we are thus looking to integrate our differential equation **restricted** to $\mathcal{T}$. For this, we need to **project the vector field** to the tangent spaces of $\mathcal{T}$.

So, our update would look like this:

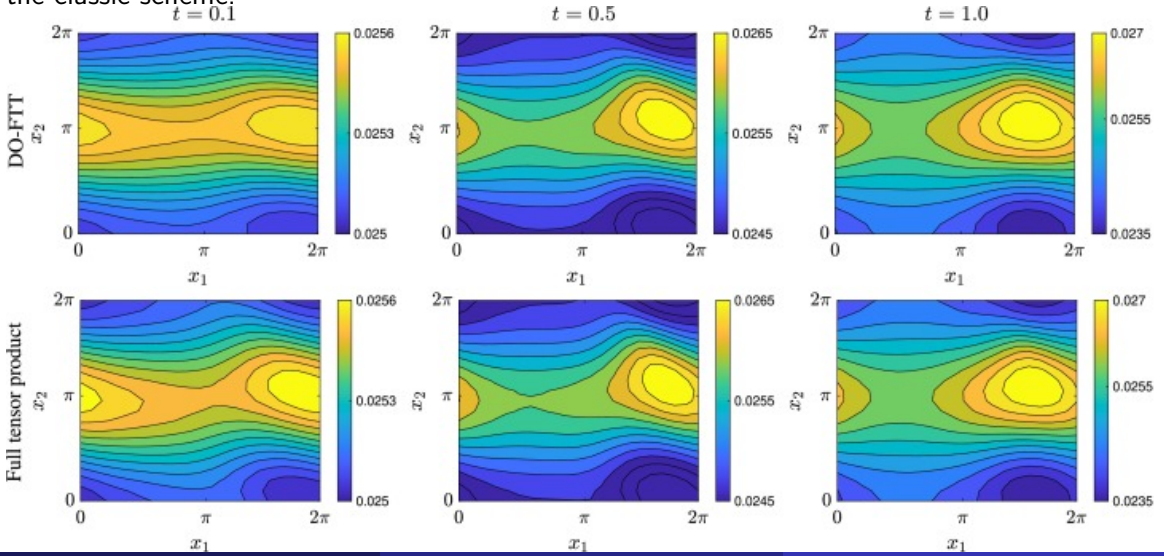
$$\hat{U}_{s+\epsilon}(x) \leftarrow \hat{U}_s(x) + \epsilon \cdot \underbrace{P_{\mathrm{T}_x\mathcal{T}}(g(\hat{U}_s(x)))}_{\text{projected tangent vector to } \mathrm{T}_x\mathcal{T}}$$

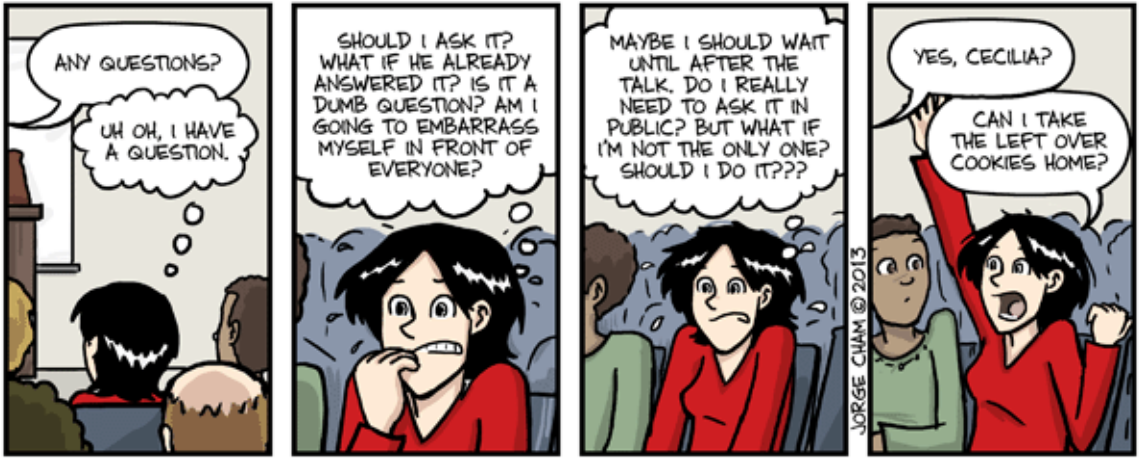
However, we cannot simply add these vectors!



In case of tensor trains, a suitable **retraction operator** R is TT-SVD.

Using this type of integration scheme, Dektor and Venturi (2021) computed the following snapshots of the approximate solution of the Fokker–Planck equation and compared them to the classic scheme:




ANY QUESTIONS?
UH OH, I HAVE A QUESTION.
SHOULD I ASK IT? WHAT IF HE ALREADY ANSWERED IT? IS IT A DUMB QUESTION? AM I GOING TO EMBARRASS MYSELF IN FRONT OF EVERYONE?
MAYBE I SHOULD WAIT UNTIL AFTER THE TALK. DO I REALLY NEED TO ASK IT IN PUBLIC? BUT WHAT IF I'M NOT THE ONLY ONE? SHOULD I DO IT???
YES, CECILIA?
CAN I TAKE THE LEFT OVER COOKIES HOME?
JORGE CHAM © 2013
WWW.PHDCOMICS.COM

Overview

1. Introduction (5')
2. Tensor network notation* (45')
   - Definition
   - Examples
   - SVD as a pair of scissors
3. Tensor trains decomposition (35')
4. Application: dynamic tensor approximation (20')
5. Conclusions

Conclusions

The tensor trains decomposition is particularly well-adapted for approximating high-dimensional tensors that represent (tensorized) solutions of physical systems modeled for example by differential or integral equations.

References

- Dektor, Venturi, *Dynamic tensor approximation of high-dimensional nonlinear PDEs*, J. Comput. Phys., 2021.
- Dolgov, Khoromskij, Litvinenko, Matthies, *Polynomial Chaos Expansion of Random Coefficients and the Solution of Stochastic Partial Differential Equations in the Tensor Train Format*, SIAM/ASA J. Uncertain, 2015.
- Dolgov, Khoromskij, Oseledets, *Fast Solution of Parabolic Problems in the Tensor Train/Quantized Tensor Train Format with Initial Application to the Fokker–Planck Equation*, SIAM J. Sci. Comput., 2012.
- Grasedyck, *Hierarchical singular value decomposition of tensors*, SIAM J. Matrix Anal. Appl., 2010.
- Grasedyck, Kluge, Krämer, *Variants of Alternating Least Squares Tensor Completion in the Tensor Train Format*, SIAM J. Sci. Comput., 2015.
- Greub, *Multilinear Algebra*, 2nd ed., Springer, 1978.
- Hackbusch, *Tensor Spaces and Numerical Tensor Calculus*, Springer, 2012.

- Holtz, Rohwedder, Schneider, *On manifolds of tensors of fixed TT-rank*, Numer. Math., 2012.
- Landsberg, *Tensors: Geometry and Applications*, AMS, 2012.
- Lubich, Rohwedder, Schneider, Vandereycken, *Dynamical Approximation by Hierarchical Tucker and Tensor-Train Tensors*, SIAM J. Matrix Anal. Appl., 2013.
- Novikov, Podoprikhin, Osokin, Vetrov, *Tensorizing neural networks*, NIPS'15, 2015.
- Orús, *A practical introduction to tensor networks: Matrix product states and projected entangled pair states*, Ann. Phys., 2014.
- Oseledets, *Tensor-train decomposition*, SIAM J. Sci. Comput., 2011.
- Oseledets, Dolgov, *Solution of Linear Systems and Matrix Inversion in the TT-Format*, SIAM J. Sci. Comput., 2012.
- Ruymbeek, Meerbergen, Michiels, *Subspace method for multiparameter-eigenvalue problems based on tensor-train representations*, arXiv:2012.00815, 2020.
- White, *Density matrix formulation for quantum renormalization groups*, Phys. Rev. Lett., 1992.
- Ye, Lim, *Tensor network ranks*, arXiv:1801.02662, 2018.